UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/726,470	12/02/2003	Shridhar Mukund	ADAPP223	5856

25920 7590 01/07/2009
MARTINE PENILLA & GENCARELLA, LLP
710 LAKEWAY DRIVE
SUITE 200
SUNNYVALE, CA 94085

EXAMINER	
MOLL, JESSE R	

ART UNIT	PAPER NUMBER
2181	

MAIL DATE	DELIVERY MODE
01/07/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/726,470	MUKUND ET AL.	
	Examiner	Art Unit	
	JESSE R. MOLL	2181	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 October 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 1-6, and 14-20 are rejected under 35 U.S.C. 102(b) as being anticipated by Narayan et al. (U.S. Patent No. 5,822,559), herein referred to as Narayan et al.'559.

Referring to claim 1, Narayan et al.'559 discloses, as claimed, a networking application processor (see Fig. 2), comprising: an input socket configured to receive data packets (the input data from I/O module of the system intended to be used); a memory (such as main memory of the Narayan et al.'559's system or instruction cache 204, see Fig. 2) for storing instructions (any memory can be used for storing instructions); circuitry configured to access data structures associated with a processing stage, the circuitry configured to access data structures enabling a single cycle access an operand from a memory location (such as main memory of the Narayan et al.'559's system or data cache 224, see Fig. 2 or the register between pipeline stages inherent in

any pipelined processor); an arithmetic logic unit (ALU) (inside function units 212A-212D, see Fig. 2); and circuitry for aligning operands (operand steering section; see fig. 29) to be processed by the ALU, the circuitry for aligning operands causing the operands to be aligned by a lowest significant bit (see col. 115, lines 7-10;

> *note that operands must be aligned by a least significant bit in order to perform calculations. For example, if two numbers are to be added [56 and 1985, or 111000 and 11111000001], the numbers inherently must be aligned by the least significant digit.*

```
       111000   (56)
+ 11111000001 (1985)
  11111111001 (2041)
```

> *Any other method of alignment prohibits correct addition),*

wherein the circuitry for aligning the operand supplies an extension (in the situation when the real operand size is less than the size to be processed) to the operand to allow the ALU to process different size operands (see col. 17, lines 32-42, regarding the operand size may be 8, 16, and 32 bits).

Referring to claim 14, Narayan et al.'559 discloses, as claimed, a processor capable of processing a data packet associated with a processing stage of a pipeline of processors (see col. 1, lines 15-20; each pipeline stage does processing and therefore any pipelined processor contains a "pipeline of processors") the processor comprising: a data random access memory (RAM) (such as the main memory of the Narayan et al.'559's system) configured to enable access to data structures; instruction fetch and decode circuitry (comprising such as early decode units 207A-207D and MROM and

anything else relating to decoding, see Fig. 2) configured to interpret instructions to be executed by an arithmetic logic unit (ALU) (function units 212A-212D, see Fig. 2), the instruction fetch and decode circuitry including, a read only memory (ROM) (such as portion of the MROM of the Narayan et al.'559's system comprising the microcode sequences; such as the exception routine described on col. 145, lines 43-56), the ROM configured to store code common to each processing stage associated with a pipeline of processors; a code RAM (The Section of MROM unit 209 storing MROM instructions, see Fig. 2; col. 14, lines 27-35 regarding storing MROM instructions), the code RAM configured to download code specific to the processing stage; and instruction decode circuitry (comprising such as decode units 208A-208D, see Fig. 2) configured to recognize operating instructions; execute and write back circuitry (comprising function units 212A-212D, see Fig. 2) configured to set up operands to be processed by the ALU, the execute and write back circuitry including, internal registers (inside register file 218, see Fig. 2) for defining a first and a second operand; an arithmetic logic unit (function units 212A-212D, see Fig. 2) for processing the first and second operands; and align function circuitry (instruction alignment unit 206, see Fig. 2) for aligning the first and the second operands to be processed by the ALU, the align function circuitry causing the first and the second operands to be aligned by a lowest significant bit, wherein the align function circuitry supplies an extension (in the situation when the real operand size is less than the size to be processed) to the each of the operands to allow the ALU to transparently process different size operands (see col. 17, lines 32-42, regarding the operand size may be 8, 16, and 32 bits).

As to claim 2, Narayan et al.'559 also discloses: the networking application processor of claim 1, wherein the instructions have a width of 96 bits (see the instruction set in Fig. 1 when it comprises 12 bytes=96 bits), <u>and wherein the single cycle access enables the data to be addressed and operated on in a single cycle in a single clock cycle without being placed into a register (see col. 10, last paragraph).</u>

<u>*Note that if an instruction is fetched, the data contained within that instruction is addressed (with a program counter) and operated on (read from memory). The definition of the word "operate" according to The American Heritage® Dictionary of the English Language, Fourth Edition is "to perform a function; work". Under this definition, a read (fetch) from memory is reasonably considered to be an operation.*</u>

As to claims 3 and 16, Narayan et al.'559 also discloses: the networking application processor of claim 1, wherein the different size operands are selected from the group consisting of 8 bit operands, 16 bit operands, and 32 bit operands (see col. 17, lines 32-42, regarding the operand size may be 8, 16, and 32 bits).

As to claim 4, Narayan et al.'559 also discloses: the networking application processor of claim 1, further including: an output socket for transmitting processed data; and a 64 bit bus (see such as 64-bit input bus to decode unit 0-3, see Fig. 25) connecting the input socket and the output socket.

As to claims 5 and 15, Narayan et al.'559 also discloses: the networking application processor of claim 1, wherein the extension to the operand fills each higher bit with a value (such as 0 for each higher bit for the unsigned operands).

As to claim 6, Narayan et al.'559 also discloses: the networking application processor of claim 1, wherein the operand is selected from the group consisting of a source operand, a destination operand, an immediate operand, and an internal register

operand (see col. 2, lines 5-23, and Fig. 1, regarding the instruction set format including source operand, a destination operand, an immediate operand).

As to claim 17, Narayan et al.'559 also discloses: the processor of claim 14, wherein the operating instructions wherein the operating instructions are formatted as 96 bit instructions (see the instruction set 100 in Fig. 1 when it comprises 12 bytes=96 bits), each of the 96 bit instructions including a single return bit (the bit such as end of file or record in the instruction set 100 in Fig. 1).

As to claim 18, Narayan et al.'559 also discloses: the processor of claim 14, wherein the processor is configured as a two stage pipeline for pipelining an instruction fetch and decode operation (using prefetch/predecode unit 202; and decode units 208A-208D, see Fig. 2) and an execute and write back operation (see Col. 144, lines 45-65, regarding write back operations).

As to claim 19, Narayan et al.'559 also discloses: the processor of claim 14, wherein the operating instructions include microcode configured to predict a likely direction for a branch instruction (using branch prediction unit 220, see Fig. 2).

As to claim 20, Narayan et al.'559 also discloses: the processor of claim 19, wherein no operation (NOP's) instructions are included (see such as col. 139, lines 4-5, regarding some of the instructions may be NOOP), the NOP's configured block an invalidated pre-fetched instruction.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 7-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Narayan et al.'559.

Referring to claim 7, Narayan et al.'559 discloses, as claimed, a processor (see Fig. 2), comprising: an input socket (the input data from I/O module of the system intended to be used) configured to receive data packets; a memory (such as main memory of the Narayan et al.'559's system or instruction cache 204, see Fig. 2) for storing instructions; circuitry configured to access data structures associated with a processing stage, the circuitry configured to access data structures enabling a single cycle access from a memory location (such as main memory of the Narayan et al.'559's system or data cache 224, see Fig. 2); and an arithmetic logic unit (ALU) (inside function units 212A-212D, see Fig. 2), the ALU configured to receive a first and a second operand (operand

A and operand B, see Fig. 33); the second operand being specified from an internal register (REGF, see Fig. 33).

Narayan et al.'559 discloses the claimed invention except for explicitly showing the first operand having a mask enabling the ALU to process a non-masked segment of the first operand.

However, Narayan et al.'559 shows using masks to select bytes before sending to the register file for the further use (see Col. 144, lines 59-60).

It would have been obvious to one having ordinary skill in the art at the time the invention was made to modify Narayan et al.'559's system to comprise the first operand having a mask enabling the ALU to process a non-masked segment of the first operand, as also taught by Narayan et al.'559, in order to facilitate selecting the useful bytes to be processed and saving the processing time in the Narayan et al.'559's system.

As to claim 8, Narayan et al.'559 also discloses: the instructions have a width of 96 bits (see the instruction set in Fig. 1 when it comprises 12 bytes=96 bits) as set forth in claim 2 above.

As to claim 9, Narayan et al.'559 also discloses: the processor of claim 7, wherein each of the instructions include a loadback feature enables random accesses to one of a source indirect register or a destination indirect register through indirect addressing (see Fig. 1 and col. 2, lines 5-22 regarding such as adding the displacement value to the content of a register to form a memory location).

As to claim 10, Narayan et al.'559 also discloses: the processor of claim 7, wherein the mask is associated with an immediate value (see Fig. 1, the instruction set comprising the immediate field) of the first operand.

As to claim 11, Narayan et al.'559 also discloses: first and the second operands are associated with a size selected from the group consisting of 8 bit operands, 16 bit operands, and 32 bit operands (see col. 17, lines 32-42, regarding the operand size may be 8, 16, and 32 bits) as set forth in claim 3 above.

As to claim 12, Narayan et al.'559 also discloses: the processor of claim 7, wherein the first operand is selected from the group consisting of a source operand, a destination operand, an immediate operand, and an internal register operand (see col. 2, lines 5-23, and Fig. 1, regarding the instruction set format including source operand, a destination operand, an immediate operand) as set forth in claim 6 above.

As to claim 13, Narayan et al.'559 also discloses: the method of claim 7, wherein the memory location (such as main memory of the Narayan et al.'559's system or data cache 224, see Fig. 2) is a static random access memory (SRAM).

Response to Arguments

5. Applicant's arguments filed 21 September 2006 have been fully considered but they are not persuasive.

6. Applicant states:

> The Office has failed to analyze Applicant's claims as a whole, instead breaking- down rejections into incoherent pieces that poorly show how the prior art arguably teaches the featured claims. For example, claim 1 defines an input socket configured to receive data packets. The Office has asserted that this feature is anticipated by Narayan because "the input data from I/O module of the system intended to be used." While this statement may be true, this statement offers absolutely no information on how Narayan anticipates Applicants' claims, and the Office's rejection is improper.

Examiner disagrees. The limitation "an input socket configure to receive data packets" merely requires a socket (any hardware for receiving data) which is able to receive data packets (any group of data). Inherently, any processor is going to send and receive data. Specifically, Narayan must perform I/O operations. These data received by the processor must be input into the processor with some sort of hardware. This hardware is "an input socket configure to receive data packets". Although the limitation is inherent in any data processor, Examiner gives a specific example of why it is in the system of Narayan for completeness and clarity.

7. Applicant states:

> Additionally, claim 1 defines a networking application processor comprising a memory for storing instructions. The Office has asserted that this feature is anticipated by Narayan in "main memory of the Narayan ... system or instruction cache 204, see Fig. 2." Applicants respectfully disagree. The person skilled in the art will readily appreciate that main memory does not reside in a processor. In fact, Fig. 2 of Narayan describing a processor does interface with the memory subsystem, therefore the main memory does not reside in the processor. The Office has chosen to ignore the claim as a whole, and the Office's rejection is improper because main memory does not reside in the processor.

Examiner disagrees. The term "processor" merely defines something that processes data. For example, a chip with two processing cores is a processor. Each core within the chip is also a processor. The same can be said about the entire computing system (including the main memory). Similarly, a system can comprise many other systems. Just because Narayan describes one part of the system as a

processor does not mean that a superset of the system is not also a processor. Additionally, the instruction cache is held within the processor as described in figure 2. Also, inherently, instructions traveling through a pipeline must be stored somewhere (intra-stage registers, wires, instruction buffers, etc.)

8. Applicant states:

> Further, claim 1 defines circuitry configured to access data structures associated with a processing stage, the circuitry configured to access data structures enabling a single cycle access of an operand from a memory location (emphasis added). The Office has not even offered an explanation of how Narayan teaches the circuitry configured to access data structures, and has merely pointed to where Narayan supposedly teaches a memory location. Further yet, the Office has not put forward either an explanation on how Narayan anticipates a single cycle access of an operand in claim 1.

Examiner disagrees. Firstly, all hardware which performs a function contains circuitry (transistors or any other implementation of logic). The term "single cycle" is broad and merely requires the access be performed in a certain timer period. Inherently, since operands are read from a memory, they must be accessed in a certain time period. Additionally, the limitation only requires the operand be accessed in one cycle. Inherently, there is a cycle in which the data is accessed (since there is no limitation reciting a requirement that the operand be read the same clock cycle that it is requested). Further, it is naive to say that nowhere in the pipelined system of Narayan is an operand read in one clock cycle). In any pipelined processor, there are registers holding information between every stage. In one clock cycle, information is passed to the next stage (including operands). The citations in the rejection are only examples and the prior art reference must be considered as a whole.

9. Applicant states:

> In addition, claim 1 defines circuitry for aligning operands to be processed by the ALU, the circuitry for aligning operands causing the operands to be aligned by a lowest -~ significant bit, wherein the circuitry for aligning the operands supplies an extension to the operands to allow the ALU to process different size operands (emphasis added). Instead of teaching how Narayan teaches this limitation, the Office has instead decided to explain how Narayan teaches a different limitation, by changing the language of the claim and instead describing how Narayan teaches "causing the operand" (in singular) "to be aligned," and "circuitry for aligning the operand" (in singular).

Examiner disagrees. The lack of an "s" on operands is clearly a typographic error. The two operands (1985 and 56 in the example shown) are clearly aligned to each other therefore two operand**s** are aligned. Additionally, any circuitry **for** aligning (as claimed) one operand can clearly align more than one. It would be foolish to assume that only one instruction would flow through a processor. Additionally, the claim language does not require the operands to be aligned, it merely requires circuitry that is able to align operands.

10. Applicant states:

> Under a 102 rejection, the Office must state how all of the claim limitations are anticipated by the prior art. Changing the Applicants' claim language is inappropriate and misleading. Even assuming arguendo that Narayan taught circuitry of aligning the operand, this does not anticipate circuitry for aligning the operands, as claimed by Applicants. Applicants respectfully request that the Office clearly explains how the potential prior art teaches Applicants claims using the exact language of the claims, or using obviousness arguments to explain the differences.

Examiner disagrees. As stated above, not only does the prior art teach aligning 2 operands, the claim merely requires circuitry capable of aligning multiple operands (which is inherent of any circuitry that can align 1 operand).

11. Applicant states:

> Applicants respectfully disagree. Narayan teaches an instruction processing pipeline within the microprocessor, which nowhere suggests a pipeline of processors. Suggesting that any pipelined processor contains a pipeline of processors is conclusory and unsubstantiated. The Office is offering a twisted play on words, which any reasonable person skilled in the art would consider inappropriate. Not everything in a system that performs a function is a processor. For example, memory performs a function of storing data, yet a person skilled in the art would never call a memory a processor. The Office has failed to show how Narayan teaches a processor capable of processing a data packet associated with a processing stage of a pipeline of processors, and the Office's rejection is improper.

Examiner disagrees. The definition of processor is "A part of a computer, such as the central processing unit, that performs calculations or other manipulations of data. " (processor. (n.d.). WordNet® 3.0. Retrieved December 23, 2008, from Dictionary.com website). Clearly, a pipeline stage falls within this definition. Similarly, if an unduly narrow definition of processor is used, only the entire system shown in Applicants' figure 3 can be considered a processor instead of each individual processing unit. In much the same way as applicants' system, the system of Narayan passes data through the pipeline. Each stage (equivalent to each of Applicant's processors) operates on the data and passes it to the next stage. A processing stage is by definition, a processor, not because of "a twisted play on words" as applicant alleges.

12. Applicant states:

> Further, claim 14 defines a processor comprising a data random access memory (RAM) configured to enable access to data structures. The Office has asserted that this featured is taught by Narayan "as the main memory." Applicants respectfully disagree. As previously described with respect to claim 1, main memory is not located in the processor, and the Office's rejection is improper.

Examiner disagrees. As stated above, the main memory is located within the processor. Additionally, the instruction cache is also RAM.

13. Applicant states:

> Additionally, claim 14 defines instruction fetch and decode circuitry configured to interpret instructions to be executed by an arithmetic logic unit, which according to the Office is taught by Narayan in 'early decode units 207A-207D and MROM and anything else related to decoding" (emphasis added). Applicants respectfully disagree. Applicants request that the Office clearly explain how the prior art teaches the claimed limitations. The assertion "and anything else" is overly broad and vague. Applicants respectfully request that the Office points out where each of the specific limitations recited in the rejected claims is found in the prior art relied on in the rejection without mischaracterizing the references. See MPEP, "the examiner should set forth in the Office action ...(A) the relevant teachings of the prior art relied upon, preferably with reference to the relevant column or page number(s) and line number(s) where appropriate" (MPEP 706.02(j)-emphasis added).

Inherently, Narayan must contain fetch and decode circuitry because instructions are fetched and decoded. Clearly, any circuitry within the system of Narayan which fetches and decodes would fall within this limitation. Examiner described the decode units and MROM as examples of the circuitry for convience. If more examples are required, these units also fall within the scope of the limitation: Instruction Alignment Unit 206, Early Decode Units 207A-207D, the pipeline stage register inherently between Early Decode Units and Decode Units, Prefetch/Predecode Unit 202, Branch Prediction Unit 220, Fetch Control Unit 266, Instruction Storage Array 250, Instruction Group Control 260, etc… These are many examples showing that instruction fetch and decode circuitry exists. For simplicity, Examiner will restate that limitation is anticipated by all circuitry used for fetching and decoding.

14. Applicant states:

> Further, claim 14 defines that the instruction fetch and decode circuitry includes a read only memory (ROM), the ROM configured to store code common to each processing stage associated with a pipeline of processors. The Office has asserted that the ROM is anticipated by Narayan in "portion of the MROM" (emphasis added). Applicants respectfully disagree. The Office has once more used vague language such as "portion" that fails to explain how the prior art teaches Applicants' claims. Moreover, Narayan teaches that "MROM unit 209 parses and serializes the instruction into a subset of defined fast path instructions to effectuate a desired operation" (col. 6, lines 29-31). Thus, a MROM does not suggest a ROM, although the names sound similar, because MROM unit parses and serializes instructions, and a Read Only Memory stores bits and does not parse and serialize instructions. Thus, Narayan does not teach that the instruction fetch and decode circuitry includes a read only memory (ROM), the ROM configured to store code common to each processing stage associated with a pipeline of processors.

As stated in the previous office action, the limitation is anticipated by "portion of the MROM of the Narayan et al.'559's system comprising the microcode". MROM units decode instructions and using the table output micro-operations using a lookup table. That table is a ROM.

15. Regarding claim 2, Examiner disagrees. The term "operated on" is extremely broad. There is nothing in the claim limiting how the data need be operated on. Simply pulling the instruction (fetching) is operating on it. Narayan clearly shows that the instruction fetch occurs in 1 clock cycle. Examiner suggest adding limitations to clearly define how the data is operated on in order to more specifically claim the invention. Instructions definitely are data structures. The opcode clearly determines how the operands will be manipulated. Additionally, claim 2 does not require a data structure: "enables **the data** to be..."

16. Regarding the rejection under USC 103, see above regarding the term "single cycle". Additionally, the claim merely requires and "internal register" and does not specify what it is internal to. The register is clearly internal to the processor.

Conclusion

17. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

18. Any inquiry concerning this communication or earlier communications from the examiner should be directed to JESSE R. MOLL whose telephone number is (571)272-2703. The examiner can normally be reached on M-F 10:00 am - 6:30 pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Alford Kindred can be reached on (571)272-4037. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Jesse R Moll
Examiner
Art Unit 2181

/J. R. M./
Examiner, Art Unit 2181

/Niketa I. Patel/
Primary Examiner, Art Unit 2181